DOCKET NO.: MSFT-1740/301921.01
Application No.: 10/600,066
Office Action Dated: August 18, 2006

PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Previously Presented)   A method for debugging an object model comprising:

exposing a set of breakpoints through a debugging interface of an execution environment, wherein a breakpoint of the set of breakpoints is associated with an object comprising a pluggable component created by the execution environment wherein the pluggable component communicates a description of the breakpoint to the execution environment;

checking a status of the breakpoint at predetermined intervals; and

responding to a request for suspend and resume.

2. (Original)   The method as recited in claim 1, further comprising communicating between a package deployment component and a pluggable component.

3. (Original)   The method as recited in claim 1, further comprising receiving input for processing by the debugging interface, the input indicative of instructions to execute or deploy a package.

4. (Original)   The method as recited in claim 1, further comprising setting the breakpoint in at least one task.

5. (Original)   The method as recited in claim 4, further comprising encountering the set breakpoint by the task during task execution.

6. (Original)   The method as recited in claim 5, further comprising communicating the breakpoint to the debugging interface.

7. (Previously Presented)   The method as recited in claim 6, further comprising analyzing by the debugging interface states of the task prior to encountering the set breakpoint.

DOCKET NO.: MSFT-1740/301921.01
Application No.: 10/600,066
Office Action Dated: August 18, 2006

PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116

8.   (Original)   The method as recited in claim 7, further comprising resuming the task by the debugging interface.

9.   (Currently Amended) A computer readable <u>storage</u> medium having instructions to instruct a computer to:

      expose a set of breakpoints through a debugging interface of an execution environment, wherein a breakpoint of the set of breakpoints is associated with an object comprising a pluggable component created by the execution environment wherein the pluggable component communicates a description of the breakpoint to the execution environment;

      check a status of the set of breakpoints at predetermined intervals; and

      respond to a request for suspend and resume.

10.   (Previously Presented)   A method for debugging an object model, comprising:

      providing an interface manager that communicates with one or more components of the object model to provide a description of a breakpoint of a set of breakpoints, the breakpoint associated with an object comprising a pluggable component created by an execution environment;

      determining the location of the breakpoint;

      executing a runtime to encounter the breakpoint.

11.   (Original)   The method as recited in claim 10, further comprising executing a package.

12.   (Previously Presented)   The method as recited in claim 11, further comprising executing at least one task resulting from the package execution.

13.   (Original)   The method as recited in claim 10, wherein the determining step comprises setting the breakpoint by the interface manager.

14.   (Original)   The method as recited in claim 10, further comprising suspending an object model component containing a breakpoint upon encountering the breakpoint.

15.   (Original)   The method as recited in claim 14, further comprising analyzing the object model components while suspended to determine if a runtime problem exists.

16.   (Original)   The method as recited in claim 14, further comprising resuming a suspended object model component.

17.   (Currently Amended) A computer readable storage medium having computer readable instructions to instruct a computer to

        provide an interface manager that communicates with one or more components of an object model;

        determine a location of a breakpoint, wherein the breakpoint is associated with an object comprising a pluggable component created by an execution environment wherein the pluggable component communicates a description of the breakpoint to the execution environment;

        execute a runtime to encounter the breakpoint.

18.   (Previously Presented)   A system to debug breakpoints in pluggable components comprising:

        a debugging interface, the debugging interface capable of communicating with the pluggable components during run time to observe component behavior and to control components, wherein a pluggable component comprises an object created by the execution environment, the pluggable component associated with a breakpoint, wherein the pluggable component communicates a description of the breakpoint to the execution environment; and

        breakpoints, the breakpoints being set in the pluggable components such that during run-time when a break point is encountered, the debugging interface is capable of suspending and/or resuming operations of the pluggable components to observe pluggable component operations.

DOCKET NO.: MSFT-1740/301921.01
Application No.: 10/600,066
Office Action Dated: August 18, 2006

PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116

19.   (Original)   The system as recited in claim 18, wherein the debugging interface sets the breakpoints in the pluggable components.

20.   (Previously Presented)   The system as recited in claim 19, wherein the debugging interface displays states of the pluggable components during run-time and during suspension.